REMARKS

Claims 1-20 are now pending in the application. Applicants have amended Claims 1 and 10 to include a master computer that clones properties of a first I/O device in order to configure a second I/O device. The Examiner is respectfully requested to reconsider and withdraw the rejections in view of the amendments and remarks contained herein.

REJECTION UNDER 35 U.S.C. § 112

Applicants respectfully traverse the rejection of Claims 1-20 under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement.

Referring to Claims 1 and 10, Applicants have amended Claims 1 and 10 to include cloning of properties of I/O devices instead of cloning of I/O devices. Therefore, Applicants believe that the rejection of Claims 1 and 10 under 35 U.S.C. § 112, first paragraph, is now moot.

Claims 2-9 and Claims 11-20 depend directly or indirectly from Claims 1 and 10, respectively. Therefore, Applicants believe the rejection of Claims 2-9 and 11-20 under 35 U.S.C. § 112, first paragraph, is also now moot.

REJECTION UNDER 35 U.S.C. § 103

Applicants respectfully traverse the rejection of Claims 1-20 under 35 U.S.C. § 103(a) as being unpatentable over Weinhofer (U.S. Pat. No. 6,442,442) in view of Stine (U.S. Pat. No. 6,466,827).

Referring to Claim 1, Weinhofer does not show, teach, or suggest a master computer that clones properties of a first I/O device that is connected to a first network including one of attributes, parameters, and operations in order to configure a second I/O device that is subsequently connected to the first network.

Weinhofer teaches a system-level data flow programming interface for a multi-axis industrial control system. The industrial control system includes one or more programmable controller systems that are connected to a network (col. 5, line 16). A plurality of input/output devices are controlled by the programmable controllers. A programming interface is used to create a user program that controls the industrial control system (col. 6, line 5). The programming interface includes icons that represent the input/output devices. Connection lines between the icons represent the physical relationships between the input and output devices (col. 6, line 17).

Weinhofer teaches that the icons are preferably objects in an object-oriented programming environment (col. 6, line 39). However, the programmable controller does not clone properties of a first input/output in order to configure a second input/output device, as required by the claims. The icons are made available to a user in a workspace. The number, type, and interconnection of the icons is determined by the user (col. 6, line 21). Therefore, icons that are inserted into the programming environment do not necessarily represent actual input/output devices.

Weinhofer teaches that a user may click on a icon in order to create a new instance of the icon. When the icons are objects in an object-oriented environment, the new instance of the icon assumes all of the attributes associated with the particular class of objects to which the original icon belonged (col. 10, line 20). However, the new

icon does not assume all of the properties of the original icon. For example, the new icon does not inherit properties that are specific to the original icon and not representative of the class of objects to which the original icon belonged. Therefore, desirable attributes, parameters, and/or operations of the original icon that are specific to the original icon may need to be re-entered manually in order to apply them to the new icon. For example, the new instance of the icon may not inherit specific current and voltage limit attributes of the original icon.

Stine does not remedy the shortcomings of Weinhofer. Stine teaches an industrial control system that employs relay ladder objects, which are associated with input/output devices. The relay ladder objects are pre-stored in a programmable logic controller and invoked when needed for a particular application as opposed to dynamically allocating memory for a relay ladder object when the relay ladder object is instanced (Abstract). The pre-stored relay ladder objects are also assigned pre-allocated variable addresses in order to limit a maximum memory usage.

The programmable logic controller does not clone properties of a first input/output device in order to configure a second input/output device, as required by the claims. The programmable logic controller pre-configures multiple relay ladder objects that belong to the same class. Therefore, the relay ladder objects belonging to the same class are initially identical. However, the relay ladder objects have different properties after the relay ladder objects are configured for operation of the specific input/output devices to which they are assigned. For example, in the system taught by Stine, it may be time consuming to configure the properties of a fist relay ladder object

in order to prepare the pre-configured relay ladder object for operation of a first input/output device.

In the event that a second input/output device is substantially similar to the first input/output device, a user is unable to clone the properties of the first relay ladder object in order to configure a second relay ladder object. The user is required to revert to a new copy of the pre-configured relay ladder object belonging to the same class and to reconfigure the relay ladder object from that point forward. Therefore, pre-configuring multiple copies of relay ladder objects that belong to the same class as taught by Stine is analogous to creating new instances of an existing icon in the system taught by Weinhofer.

Applicants teach a system for cloning input/output (I/O) devices connected to a network of an industrial control system. In paragraph **[0005]** of the application, Applicants state that an industrial process may include hundreds of I/O devices that each include over 100 attributes. The object oriented model taught by Applicants automatically clones I/O devices including the attributes and/or operating parameters of the I/O devices. For example, a user may configure a first I/O device with a set of attributes. The user may then copy the first I/O device to create a second I/O device. Attribute values of the second I/O device may then be modified if desired. Therefore, the system taught by Applicants significant reduces time and computing resource usage.

Claims 2-9 depend directly or indirectly from Claim 1 and are allowable over Weinhofer and Stine for the same reasons.

Referring to Claim 10, Weinhofer does not show, teach, or suggest a master computer that clones properties of a first I/O device that is connected to one of first and second networks including one of attributes parameters, and operations in order to configure a second I/O device that is subsequently connected to the other of said first and second networks.

The arguments made above with respect to Claim 1 are equally applicable to Claim 10. The programmable controller taught by Weinhofer does not clone properties of a first input/output in order to configure a second input/output device. Icons that are inserted into the programming environment do not necessarily represent actual input/output devices. New instances of icons do not assume all of the properties of the original icon. Therefore, desirable attributes, parameters, and/or operations of the original icon that are specific to the original icon may need to be re-entered manually in order to apply them to the new icon.

Stine does not remedy the shortcomings of Weinhofer. The programmable logic controller taught by Stine does not clone properties of a first input/output device in order to configure a second input/output device. The programmable logic controller pre-configures multiple relay ladder objects that belong to the same class. The relay ladder objects have different properties after the relay ladder objects are configured for operation of the specific input/output devices to which they are assigned. Pre-configuring multiple copies of relay ladder objects that belong to the same class as taught by Stine is analogous to creating new instances of an existing icon in the system taught by Weinhofer.

CONCLUSION

It is believed that all of the stated grounds of rejection have been properly traversed, accommodated, or rendered moot. Applicants therefore respectfully request that the Examiner reconsider and withdraw all presently outstanding rejections. It is believed that a full and complete response has been made to the outstanding Office Action, and as such, the present application is in condition for allowance. Thus, prompt and favorable consideration of this amendment is respectfully requested. If the Examiner believes that personal communication will expedite prosecution of this application, the Examiner is invited to telephone the undersigned at (248) 641-1600.

Respectfully submitted,

Dated: 4/11/05

By: _____
Michael D. Wiggins
Reg. No. 34,754

HARNESS, DICKEY & PIERCE, P.L.C.
P.O. Box 828
Bloomfield Hills, Michigan 48303
(248) 641-1600

MDW/wmt